

Torres de colores. Versión 2

(Examen febrero 2017, sesión 2, ejercicio 3, simplificado) Laura quiere construir una torre con piezas de colores. En su juego de construcciones hay piezas azules, rojas y verdes, de cada una de las cuales tiene tantas piezas como quiera. Quiere construir una torre que contenga $n \geq 1$ piezas en total cada una encima de la anterior. No le gusta el color verde, así que nunca coloca dos piezas verdes juntas mientras se va construyendo la torre. Además, como el color rojo es su favorito, las torres que construye siempre tienen en la parte inferior una pieza roja.

Implementar un algoritmo que muestre todas las formas posibles que tiene de construir una torre de la altura deseada cumpliendo con las restricciones mencionadas.

Requisitos de implementación.

Añadir a la versión anterior una condición *esValida* que comprueba que no se colocan dos piezas verdes seguidas. La torre roja en la base se debe colocar en la función *resuelveCaso*, antes de llamar a la función recursiva y entrar en la función recursiva en la etapa 1.

Entrada

La entrada que espera el corrector automático consta de una serie de casos de prueba y acabará cuando se introduzca una línea con un cero. Cada caso de prueba se escribe en una línea y consta de 1 entero mayor que cero y representa la altura de la torre.

Salida

Para cada caso de prueba se escriben todas las posibles torres, una en cada línea ordenadas por orden lexicográfico y separando cada par de colores por un espacio. Cada caso termina con una línea en blanco.

Entrada de ejemplo

```
1
2
3
0
```

Salida de ejemplo

```
rojo
rojo azul
rojo rojo
rojo verde

rojo azul azul
rojo azul rojo
rojo azul verde
rojo rojo azul
rojo rojo rojo
rojo rojo verde
rojo verde azul
rojo verde rojo
```